INFORMATION AND CONTROL **26**, 194–197 (1974)

Note

A Note on Probabilistic Input–Output Relations*

S. RAO KOSARAJU

The Johns Hopkins University, Baltimore, Maryland 21218

Paz (1971a, p. 65) poses the following open problem: Given a recursively computable probabilistic input–output relation p, formulate a decision procedure for ascertaining whether p is of finite rank, or prove that the problem is not decidable. In this note, we prove that the problem is not decidable.

1. INTRODUCTION

We follow the terminology of Paz.

DEFINITION 1. A *probabilistic input–output relation* is a function $p(v \mid u)$ whose domain is the set of pairs (v, u) of output–input sequences (of equal length) over respective finite input and output alphabets X and Y, whose range is the interval $[0, 1]$, and subject to the restrictions:

(i) $p(\Lambda \mid \Lambda) = 1$, where Λ is the null string, and

(ii) $\sum_y p(vy \mid ux) = p(v \mid u)$ for all $x \in X$, the summation is over all $y \in Y$.

Throughout this paper the term "relation" refers to a probabilistic input–output relation.

Notation. $\mathscr{T}(X, Y)$ denotes the class of all relations over the input and output alphabets X and Y.

DEFINITION 2. Let (u_1, v_1), (u_2, v_2),..., (u_n, v_n), (u_1', v_1'), (u_2', v_2'),..., (u_n', v_n') be a set of $2n$ pairs of sequences such that length of u_i = length of v_i, and length of u_i' = length of v_i' for $i = 1,..., n$, and let $p \in \mathscr{T}(X, Y)$ be a relation. The matrix $P = [p(v_i v_j' \mid u_i u_j')]$ is then called a *compound sequence matrix.*

* Supported by National Bureau of Standards under Contract No. FL13-3-35999.

DEFINITION 3. The rank $r(p)$ of a relation p is the maximum among the ranks of all compound sequence matrices which can be formed from p, or $+\infty$ if no such maximum exists.

DEFINITION 4. In any recursive enumeration (Rogers, 1967) of Turing machines, let TM_i denote the ith Turing machine. Let φ_i be the partial recursive function computed by TM_i.

2. MAIN RESULT

LEMMA 1. *For* $X = \{\alpha\}$ *and* $Y = \{a, b\}$, *let* p *be the relation*

$$p(\Lambda \mid \Lambda) = 1,$$

and, for $j \geqslant 1$,

$$p(y \mid \alpha^j) = \begin{cases} 1 & \textit{if} \quad y = a^j, \\ 0 & \textit{otherwise}. \end{cases}$$

Then the rank of p *is* 1.

Proof. Trivial. Q.E.D.

For every i, let us define a relation $p_i \in \mathscr{T}(\{\alpha\}, \{a, b\})$ as follows:

(1) $p_i(\Lambda \mid \Lambda) = 1$, and

(2a) if TM_i does not halt on input i (i.e., $\varphi_i(i)$ is undefined), then for $k \geqslant 1$

$$p_i(y \mid \alpha^k) = \begin{cases} 1 & \text{if} \quad y = a^k, \\ 0 & \text{otherwise}; \end{cases}$$

(2b) if TM_i halts on input i on the mth step, then for $k \geqslant 1$

$$p_i(y \mid \alpha^k) = \begin{cases} 1 \text{ if } 1 \leqslant k \leqslant m \text{ and } y = a^k, \text{ or} \\ \quad k = j(2m + j + 1)/2 + j',\, y = a^m b a^{m+1} b \cdots b a^{m+j-1} b a^{j'} \\ \quad \text{where } j \geqslant 1 \text{ and } 0 \leqslant j' \leqslant m + j, \\ 0 \text{ otherwise}. \end{cases}$$

LEMMA 2. *If* TM_i *does not halt on input* i, *then* p_i *has rank* 1.

Proof. From the definition of p_i and Lemma 1. Q.E.D.

LEMMA 3. *If* TM_i *halts on input* i, *then* p_i *has rank* $+\infty$.

Proof. Let TM_i halt on input i on the mth step. For any $n \geqslant 1$ and $j = 1, 2, \ldots, n$ let

$$(u_j, v_j) = (\alpha^{j(2m+j+1)/2}, a^m b a^{m+1} b \cdots b a^{m+j-1} b)$$

and

$$(u_j', v_j') = (\alpha^{m+j+1}, a^{m+j} b).$$

Then $[p_i(v_j v_k' \mid u_j u_k')]$ is the $n \times n$ identity matrix, and hence of rank n. Thus there is no bound on the ranks of the compound sequence matrices that can be formed from p_i. Q.E.D.

COROLLARY 1. *The relation p_i has finite rank if and only if TM_i does not halt on input i.*

Proof. Direct from Lemmas 2 and 3. Q.E.D.

LEMMA 4. *For every i, p_i is recursively computable, and the description of the Turing machine for p_i is recursively computable from i.*

Proof. We describe how to compute $p_i(y \mid \alpha^k)$. If y is not of length k, then make $p_i(y \mid \alpha^k)$ undefined. If y is of length k, then apply the following procedure.

Enumerate TM_i and start its simulation on input i.

(a) If TM_i on input i does not halt in $k - 1$ steps, then set

$$p_i(y \mid \alpha^k) = \begin{cases} 1 & \text{if } y = a^k, \\ 0 & \text{otherwise.} \end{cases}$$

(b) If TM_i on input i halts on the mth step and $m \leqslant k - 1$, then set

$$p_i(y \mid \alpha^k) = \begin{cases} 1 & \text{if } y = a^m b a^{m+1} b \cdots b a^{m+j-1} b a^{j'}, \\ & \text{where } j \geqslant 1,\ 0 \leqslant j' \leqslant m + j, \\ 0 & \text{otherwise.} \end{cases}$$

Thus p_i is recursively computable, and the above description shows that the description of the Turing machine for p_i can be recursively computed from i (Rogers, 1967). Q.E.D.

THEOREM 1. *The problem of ascertaining whether a given recursively computable relation has finite rank is undecidable.*

Proof. For any i, define p_i as given earlier. By Corollary 1, p_i has a finite rank if and only if TM_i does not halt on input i. By Lemma 4, p_i is recursively computable, and in addition the description of the Turing machine for p_i is recursively computable from i. Hence, if the problem of ascertaining whether a given recursively computable relation has finite rank is decidable, then the halting problem is decidable. Q.E.D.

Note Added after Submission. Thanks to the referee who informed us that the reference by Paz (1971b) contains another proof of Theorem 1.

RECEIVED: February 14, 1974

REFERENCES

PAZ, A. (1971a), "Introduction to Probabilistic Automata," Academic Press, New York.

PAZ, A. (1971b), Formal series, finiteness properties and decision problems, *Ann. Acad. Sci. Fennicae Ser. A, Math.* **693**, 12.

ROGERS, H. (1967), "Theory of Recursive Functions and Effective Computability," McGraw-Hill, New York.